TaiWanese LOM (TW LOM) Annotation: Automatic Description Generation for Learning Objects

Joe Chun-Te Lee[a], Sin-Jie Haung[b], Yi-Wen Chiang[c], Chun-Chieh Liu[d], and Von-Wun Soo[e]
[a,e]Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan 30013.
[b,c,d]Institute of Information Systems and Applications, National Tsing Hua University, Hsinchu, Taiwan 30013.
[e]Department of Computer Science and Information Engineering, National University of Kaohsiung, Kaohsiung, Taiwan 800.
lct4246@gmail.com

Abstract: Owning to the great growth of e-learning objects, authorities (e.g. ADL and IEEE) have developed some metadata standards to facilitate the keyword search for various e-learning applications. However, too much fields, such as 58 blank fields in IEEE LOM, waiting for authors or annotators to fill up become an endless nightmare. In order to reach our vision of sharing and reusing valuable assets, the needs for an intelligent and automatic annotation system become more and more urgent. Among these 58 elements, it is the most difficult to extract the fittest solutions for Description, which calls for the advanced Chinese language processing technologies to generate the suitable value. We also adopted the Self-Organizing Map clustering method from Neural Networks, feature selection from Information Retrieval, and Latent Semantic Analysis from Linguistics to cope with the automatic annotation problem.

In this paper, we proposed a novel approach called Clustering Descriptor, CD, to automatically generate the description metadata in TW LOM—a Learning Object Metadata application profile in Taiwan. Then, we conducted two experiments to evaluate the annotation quality for Description data element in terms of three parameters: clustering, feature weight, and semantics. Because of the benefits from clustering and feature weight, Clustering Descriptor achieved improvement in precision rate: 6.30% (clustering) and 8.60% (clustering plus feature weight) compared with the baseline.

Keywords: Description generation, metadata annotation, summarization, TW LOM, e-learning

Introduction

As we all know, designing a suit of good-quality *learning objects* is the time-consuming and labor-intensive work. For this reason, how to facilitate the *reusability* of existing ones and create their new added values become the more and more crucial issues for many e-learning applications. In order to achieve this ultimate goal, one approach is to annotate them with certain descriptive *metadata* so that the relevant learning objects, closely matching user's needs, could be easily retrieved later on. However, to improve learning object's reusability with metadata annotation, we must create certain *e-learning standards* in advance. Hence, *IEEE (Institute of Electrical and Electronics Engineers) LOM (Learning Objects Metadata)* [27] and *ADL (Advanced Distributed Learning) SCORM (Sharable Content Object Reference Model)* [25] are developed to meet these demands.

Figure **1** introduces a simple metadata-based *LMS (Learning Management System)* architecture [13] to explain how we could use the annotated learning objects to apply to various applications.

When users login the LMS, they first register their personal information to a *user-profile management module*. Commonly, at least two basic operations are user-friendly allowed: *query* and *annotation*.

Query. Using queries in Chinese language, users can send their requests via the *query interface* to the *query-processing module*, and the LMS automatically conducts the *Chinese*

Language Processing (CLP) tasks. Then, the LMS responses the highest relevant learning objects stored in its repository to the *content aggregation module*, so as to reassemble these learning objects for further use. Finally, users could get the reusable learning objects aggregated from the existing ones.

Annotation. Either authors or annotators can annotate learning objects using the *annotation interface*. System could automatically or semi-automatically generate the suitable values accompanied with the user profile to fill up the *application profile (AP)* for *TW LOM (Taiwanese LOM)*. Thus, the annotated learning objects could be stored back to the centralized or distributed repository.

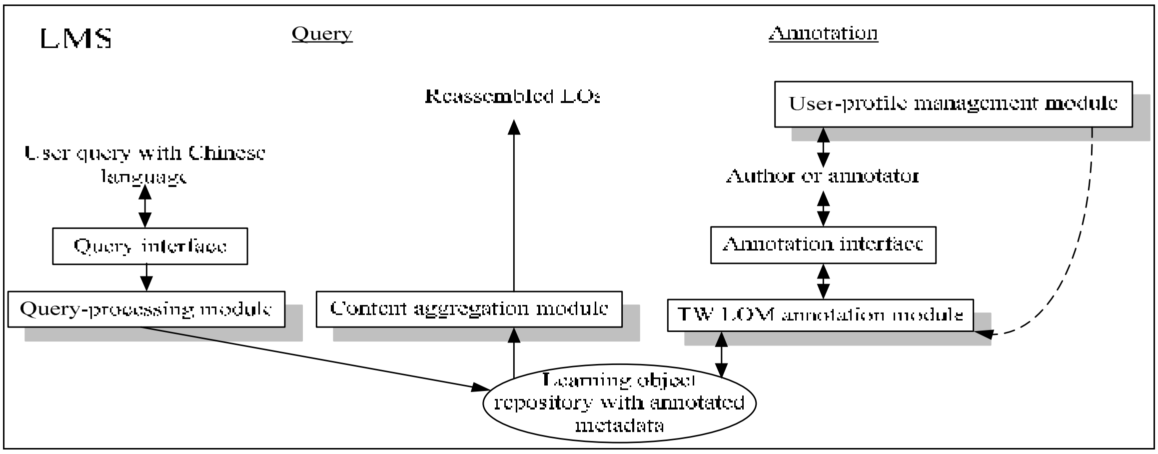

Figure 1: Simple LMS architecture to facilitate two basic learning-object operations: *query* and *annotation*.

However, it is impossible to annotate huge amount of learning objects without computer help in a short time. The annotation process is not only tedious and boring, but also prone to inconsistencies for humans—*the same annotator who annotates the same learning objects at the different time might provide different answers* [13]. Besides, for each learning object, IEEE LOM consists of fifty-eight blank fields waiting for annotation. Some are mandatory; others are optional. Their values much depend on the application profile. For a naïve author or annotator, he is easy to be stuck at certain unfamiliar fields and to eventually leave empty values for some unsure ones. Therefore, designing an intelligent and automatic annotation system without too much intervention for humans has attracted much attention for the e-learning communities and societies.

Among those fields to be annotated in TW LOM, the toughest one might be the *general description* for a learning object. An annotator must fill it up with some phrases or short sentences to highlight the learning object's content. To some extent, it is equivalent to add some *comments* summarized from the overall learning-object contents. To automate this process, after we adopt *content mining* to analyze the learning objects, many advanced technologies might still need to be involved: *Chinese language processing*, *keyword extraction*—the average performance is around 80% [13], *summarization*—the average accuracy is about 50% [7][11][23]. In this paper, we proposed a practical approach that could automatically generate a meaningful summary as a description of the TW LOM metadata.

1. Related Work

In this paper, we conducted a thorough survey from Luhn's pioneer work in 1958 [13] to the prevalent summarization technologies—summarization using the *supervised* vs. *unsupervised learning* [13][7][22], the *lexical* [2][3][5][20] vs. *semantic sentence extraction* [6][6][16][20][23], and the *conditional random fields* [19]. By definition, description is a *textual description of the content of this learning object*. Hence, we could treat this descriptive problem as mining the most significant sentences from those learning objects (corresponding to the *information retrieval (IR)* or the *text mining* research work).

[23] proposed two approaches to extract the important sentences for a summary. The first one is *corpus-based* [11]. Authors consider *sentential positions*, *keywords*, and *feature weight* as prominent constituents. They tried to formulate a good summary using those parameters. Their

second method adopted the *latent semantic analysis (LSA)* and *text relationship map (TRM)* [18] to dig out Chinese document's structures in order to find the significant candidate sentences comprising the final result.

For a certain given Chinese document [22], we segment them into sentences according to some *terminating punctuations* such as a question mark (？), an exclamation mark (！), a period (。), and so on. Then, we extract features from the previously segmented sentences based on their *mutual dependencies*, and obtain many candidate features at hand. Using the *fuzzy C-means* clustering algorithm, we obtain some clusters regarded as *subtopics* based on the extracted candidate features beforehand, and combine them to generate the final answer.

By the Chinese language processing techniques, we could also extract candidate keywords from the text easily [8]. The *semantic networks* and *context relations* are used to attach *linguistic meanings* to those keywords. We aggregate them into groups of different concepts, and select the most significant sentences by evaluating their salient features in the long run. Then, the desired summary could be successfully obtained.

With nearly half-decade evolution, new methods, on the one hand, are still under development; another important issue (evaluation among summaries), on the other hand, pays much attention by the societies—Though more and more advanced techniques have been invented, "How good is it?" frequently struggles with our mind [10]. In 90's, U.S. government, *DARPA* (*Defense Advanced Research Projects Agency*), launched the *TIPSTER Text Program*: *TIPSTER Text Summarization Evaluation (SUMMAC)* to conduct the large-scale assessment on document processing [30]. They focused on three areas: *Document Detection*, *Information Extraction*, and *Summarization*, and divided their mid-term project into three phases: *Phase I* (1991-1994), *Phase II* (April 1994-September 1996), and *Phase III* (1996) [4]. Text summarization evaluation was one of their goals at Phase III. [4] assessed fifteen research projects to approach the text summarization in the realistic scenarios.

Learning objects, analogy to the text documents in certain extent, could be regarded as an *information pool*, for instance, words, keywords, topic words, sentences, topic sentences, paragraphs, etc. If we further consider the *compression rate* of the information gain within it, we might mine various kinds of significant knowledge from the huge learning objects. Take the hottest application: *text messaging* on the cell phones for an example, because of the small display screen size, summarized headline news less than one hundred words becomes urgent demands for the service providers [21].

In this paper, we only focus on one of the prevalent summarization techniques called *sentence selection*. That is, we adopt some strategies to weight the chosen candidate sentences, and pick up the highest ranked ones to be the potential values for the Description data element in our TW LOM application profile.

2. Automatic Description Generation

We proposed a method called *Clustering Descriptor (CD)* [9][13] to automatically generate the meaningful value for the description. Why adopt the *clustering techniques* instead of others? Before answering this research question, let us look back to see how people select sentences as a description in reality. That is, *annotator's behavior problem*.

Assumption 1: In a common situation, an annotator would not pick up other relevant sentence(s) as the learning object description after he already chooses certain candidate sentence(s). That is, no *semantic redundancy* would be involved within the description.
Assumption 2: On average, we choose *three to five* sentences as the resulting description.

We attempt to refer some new ideas to improve *corpus-based* description generation. First, we pay much attention to other features than [14] does so as to find out more suitable *thematic words* involved with the discourse. Moreover, we adopt *Self-Organization Map (SOM)* from *Neural Networks* to group sentences having similar meanings into clusters [23]. Furthermore, in order to enhance clustering results, we use Latent Semantic Analysis to mine the implicit semantic meanings.

In this paper, we consider the following 4 features: *thematic word*, *position*, *TFIDF (Term Frequency and Inverse Document Frequency)*, and *title* to decide whether a sentence is an ideal candidate or not.

F_1: Thematic Word. *A critical sentence often consists of more thematic words [13].* For a sentence *s* containing n words (length is n): $word_1$, $word_2$, and $word_n$, we define its *thematic score* as F_1, where a thematic word consisting of five components is listed in Table 1.

$$F_1(\text{sentence s}) = \frac{\text{Thematic words in s}}{|s|} \quad \text{……………………………………………} (1)$$

F_2: Position. *Key sentences are often located at some significant positions.* For instance, *the first sentence in the first paragraph* usually includes many informative concepts. So, we regard those places as the higher priority than others. For each s belonging to a sentence set *S*, when we take its position i into account, we define its *positional score* as conditional probability F_2:

$$F_2(s) = P(s \in S \mid \text{Position}_i \text{ in the learning object}) \quad \text{…………………………} (2)$$

Table 1: Thematic words.

Feature		Comment
Frequency	TFIDF	● TF: term frequency ● IDF: the number of learning objects containing a specific term
Position	inTitle	Position is *title* or *heading*.
	inFirst	Position is *the first paragraph*.
Part of Speech	POS	E.g. “<Nb>”, ”<VT>”
Punctuation	inPun	E.g. “ “, < >, ’ ‘
Morphology	significance, sing	*sing(word) = g(x)*, where g(x) is a *heuristic function* about the word length; $g(x) = \begin{cases} 1, & x = 1 \\ \log 2x, & 2 \le x \le 8 \\ 3, & x > 8 \end{cases}$ [24].

F_3: TFIDF. *TFIDF can measure the semantic strength among words and learning objects.* Hence, we use it to calculate the competence for every sentence. If a sentence contains plenty semantic meanings, its score will be higher than other sentences. We call TFIDF score F_3:

$$F_3(s) = \frac{\sum_{i=1}^{n} TF \times IDF_i}{|s|} \quad \text{…………………………………………………………} (3)$$

F_4: Title Word. *A sentence is much similar to the title in semantics when it contains more title words.* F_4 formulates this definition:

$$F_4(s) = \frac{\#(\text{Title Words in s})}{|s|} \quad \text{…………………………………………………} (4)$$

After linearly combining F_1, F_2, F_3, and F_4 with certain weight: α, β, δ, and ω, we define the *score function, SF(s),* for each sentence s. The importance for s in the final description annotation highly depends on SF.

$$SF(s) = \alpha \times F_1 + \beta \times F_2 + \omega \times F_3 + \delta \times F_4 \quad \text{……………………………………} (5)$$

Definition 1: **Description**. *A description is a sentences collection containing the most significant sentences within a learning object.*

Definition 2: **The Representative Sentence in Description**. *A sentence is the representative one in the description since it gets higher score than others with four feature measurement in (5).*

In section 2.1 and 2.2, we show how *Self-Organizing Map* and *Clustering Descriptor* adapt to the learning object description generation.

2.1 Self-Organizing Map Approach

Self-Organizing Map is a *clustering algorithm* used to map a multi-dimensional dataset onto a two-dimensional map. The output results are viewed in a *Self-Organizing Map unit* (Figure 2) [29]. The advantage of Self-Organizing Map is that we don't need to decide the value of k as in the *k-means* clustering algorithm [23]. Therefore, after dividing sentences into clusters, we could pick up a representative from every cluster to become the potential value for the description.

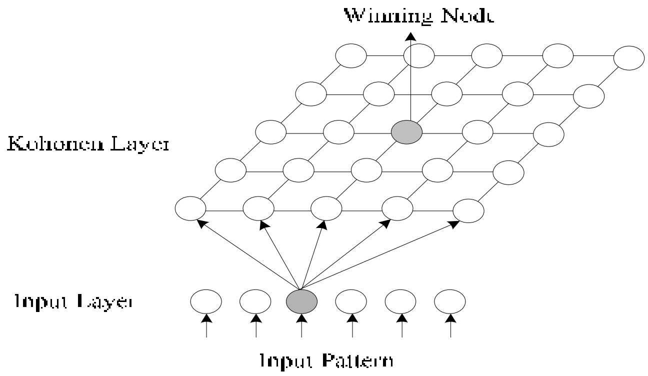

Figure 2: A Self-Organizing Map unit.

The framework (Figure 3) we proposed consists of 3 phases:

Phase 1: **Pre-processing**. Convert a learning object into a plain text file, and feed it into the *Chinese Parser* [25] for *tagging* and *segmentation*. We only concern nouns and verbs as potential concepts within sentences.

Phase 2: **Concept Clustering**. Use Self-Organizing Map to group sentences into clusters by their concepts.

Phase 3: **Sentence Selection**. Rank the cluster priority by SF (5), and find out the candidate sentences as the most possible value for the description.

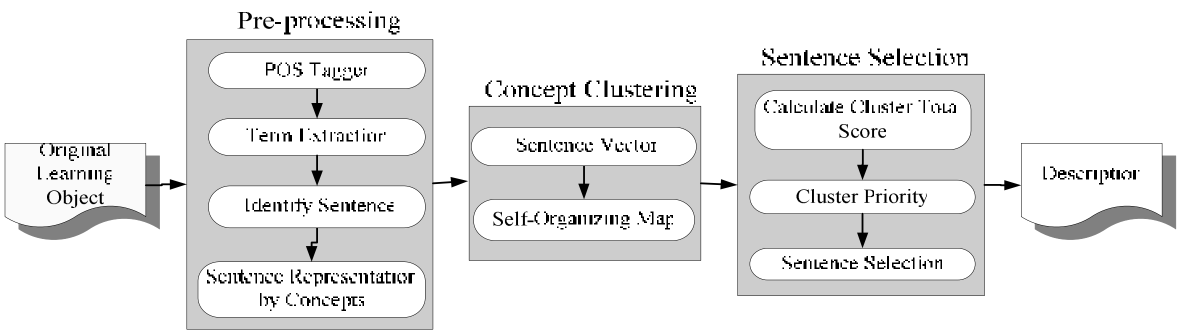

Figure 3: Framework of our Self-Organizing Map approach.

2.2 Clustering Descriptor

Latent Semantic Analysis can extract and infer relations from passages to make the clustering results more meaningful [11]. The framework of the Clustering Descriptor is shown in Figure 4. In order to elaborate more subtle semantics, we adapt *Latent Semantic Analysis model* to our problem domain. We regard the plain text file from pre-processing phase as a *word-by-sentence matrix*, *M*, and construct a new matrix, *M'*, with add-on semantics using *singular value decomposition (SVD)* and *dimension reduction*, where each column represents sentences in semantics; each row represents words in semantics. Then, we convey M' into Self-Organizing Map to gain the clustering performance.

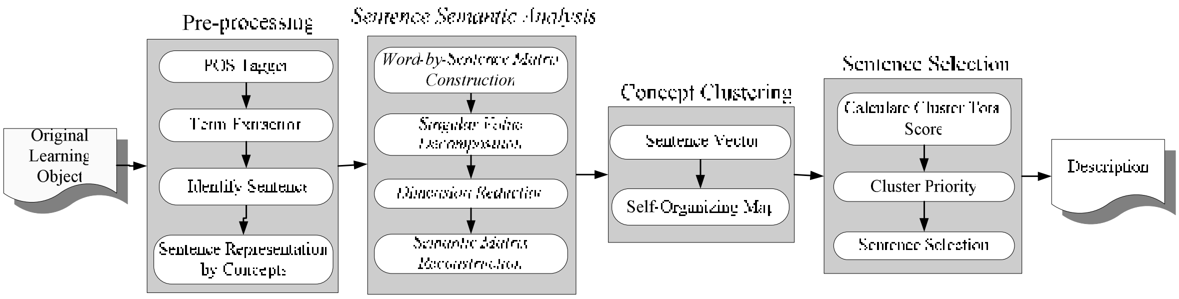

Figure 4: Framework for Clustering Descriptor.

2.3 Description Generation

In order to decide the cluster's priority, we sum up all sentential scores in each cluster. So, critical clusters would get higher scores. Besides, because title sentence is commonly comprised of the most likely semantics for the discourse, we assign those clusters containing the title sentence the highest priority.

After those complex processes, we can dig out the final descriptive sentences. Take Figure 5 as an example, we define that every description consists of three sentences. From cluster's ranking, cluster C_1, C_2, and C_3 get the highest scores while cluster C_1 contains the title sentence S1. Therefore, the final description is (S1, S5, and S4).

Figure 5: Cluster C_1, C_2, and C_3 get the highest scores among others. Because cluster C_1 contains the title sentence S1, it is our first candidate—others are selected from the rest clusters ranked by their scores. So, the final description is (S1, S5, and S4).

3. Experiments and Results

Training data. We adopt articles of earth science from [28], and choose one hundred articles as our training data. They contain nine to twenty-eight sentences (fourteen sentences in average). After cleaning them up, we ask four graduate students for help. They annotate them manually by selecting some fittest sentences. Hence, we conduct the experiments, and compare the differences between experts and our Clustering Descriptor approach.

Evaluation. We use $\text{Precision} = \frac{|A \cap B|}{A}$ and $\text{Recall} = \frac{|A \cap B|}{B}$ [23] to evaluate the results, where A denotes the number of sentences in the descriptions automatically generated by a computer; B denotes the number of sentences in the descriptions manually annotated by people. Since A and B are set to be equal in our assumption (section 2.3), we only list precision measure in the following.

Baseline. We implement two methods called *RANDOM* and *No_Clustring* as our baseline measurement. For the former, we randomly pick up descriptive sentences; for the later, we only consider four features mentioned in section two without clustering.

Experiment one: Clustering Descriptor without feature weight ($\alpha = \beta = \omega = \delta = 1$). Table 2 shows that Clustering Descriptor could actually improve the resulting performance—its precision raises **6.30%** from No_Clustering to Clustering Descriptor, and **8.31%** from Self-Organizing Map to Clustering Descriptor.

Table 2: Experiment one—Clustering Descriptor without feature weight.

	RANDOM	No_Clustering	SOM	CD
Precision	0.289398	0.584527	0.564470	**0.647564**

Experiment two: Clustering Descriptor with feature weight. We fine tuned four weighting parameters by changing one out of four at a time, and obtain an approximate optimal weight set: {5:3:4:1, 5:7:9:10, 3:3:3:5}. Table 3 shows that Clustering Descriptor is improved significantly. Its precision increases **6.30%** from No_Clustering to Clustering Descriptor, and **7.16%** from Self-Organizing Map to Clustering Descriptor.

Table 3: Experiment two—Clustering Descriptor with feature weight.

	No_Clustering	SOM	CD
α:β:ω:δ	5:3:4:1	5:7:9:10	3:3:3:5
Precision	0.607450	0.5988540	**0.670487**

Finding. The small decrease—2.01% (0.584527 - 0.564470) in experiment one and 0.86% (0.607450 – 0.5988540) in experiment two—from No_Clustering to Self-Organizing Map is because clustering benefits somehow depend on the *sentence length* in a learning object. After examining the clustered sentences again, we found an interesting phenomenon that might be depicted in Figure 6 and Figure 7. It corresponds to the "aboutness problem" [2]—*a good description highly relies on the information gain within the sentences*. Hence, we could infer that *the longer the sentences are, the higher possibility they contain the overlapped thematic words*.

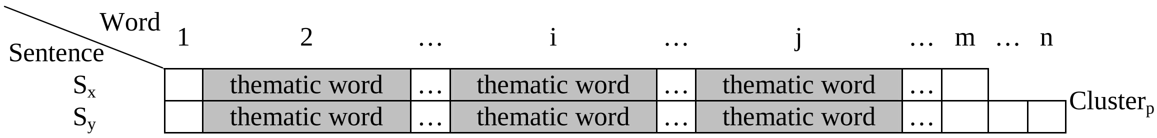

Figure 6: Longer sentences are likely grouped into the same cluster.

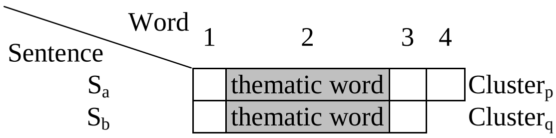

Figure 7: Shorter sentences are likely divided into different clusters.

Moreover, the net effect of *clustering plus feature weight* also shows its advantage—**8.60%** increase in precision rate: 0.584527 (no clustering and feature weight in Table 2) to 0.670487 (clustering and feature weight in Table 3).

4. Conclusions

More and more learning objects have been created nowadays. Manual annotation becomes infeasible for the annotators or authors. To reduce their burden, an intelligent and automatic annotation system is taken into account for facilitating the learning object's reusability in e-learning applications. In this paper, we propose a novel approach, Clustering Descriptor, which adopts Chinese language processing technologies (such as *tagging*, *key word extraction*, and *summarization*), *Neural Networks*, *Information Retrieval*, and *Latent Semantic Analysis* to automatically generate Description metadata. From the two experiments discussed in section three, we found that *clustering techniques* (such as k-means, and Self-Organizing Map), *feature selection* (title, thematic words, etc.), and *semantic analysis* (e.g. Latent Semantic Analysis) contribute for certain extent to the final performance. It is because our Clustering Descriptor much takes the relevant *features* and *semantics* into consideration. Therefore, it improves the precision for description generation. We have also developed automatic annotation systems [13] for other metadata fields in TW LOM that could be integrated in the future as a tool to facilitate learning object's reuse.

Acknowledgements

This study is supported by the "96 International Repository Technology Development Three Years Research Project" of the Institute for Information Industry subsidized by the Ministry of Economy Affairs of the Republic of China.

References

[1] Azzam, Saliha, Humphreys, Kevin, and Gaizauskas, Robert (1999). Using Coreference Chains for Text Summarization. *Proceedings of the ACL Workshop on Coreference and its Applications,* 77-84.

[2] Barzilay, Resina, and Elbadad, Michael (1997). Using Lexical Chains for Text Summarization. *Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97),* 10-17.

[3] Brunn, Meru, Chali, Yllias, and Pinchak, Christopher J. (2001). Text Summarization Using Lexical Chains. *Proceedings of the Document Understanding Conference,* 135-140.

[4] Gee, F. Ruth (1998), The TIPSTER Text Program Overview. *Proceedings of a workshop on Annual Meeting of the ACL,* Baltimore, Maryland, 3-5.

[5] Goldstein, Jade, Kantrowitz, Mark, Mittal, Vibhu, and Carbonell Jaime (1999). Summarizing Text Documents: Sentence Selection and Evaluation Metrics. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval,* 121-128.

[6] Gong, Yihong, and Liu, Xin (2001), Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval,* 19-25.

[7] Hu, Po, He, Tingting, and Ji, Donghong (2004). Chinese Text Summarization Based on Thematic Area Detection. *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop,* 112-119.

[8] Hu, Po, He, Tingting, Ji, Donghong, and Wang Meng (2004). A Study of Chinese Text Summarization Using Adaptive Clustering of Paragraphs. *The Fourth International Conference on Computer and Information Technology (CIT '04),* 1159-1164.

[9] Huang, Sin-Jie (2008), Using Latent Semantic Analysis and Self-Organizing Map in Chinese Text Summarization. *Hsinchu, Taiwan: Master thesis*, National Tsing Hua University.

[10] Jones, Karen Spärck (2007), Automatic Summarising: The State of the Art. *Information Processing and Management*, Volume 43, Issue 6, 1449–148.

[11] Kupiec, Julian, Pedersen, Jan, and Chen, Francine (1995). A Trainable Document Summarizer. *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, 68-73.

[12] Landauer, Thomas K., Foltz, Peter W., and Laham, Darrell (1998). An Introduction to Latent Semantic Analysis. *Discourse Processes, 25*, 259-284.

[13] Lee, Joe Chun-Te, Haung, Sin-Jie, Yang, Ting-Hao, Soo, Von-Wun, and Lee, Chen-Yu (2008). *Intelligent Annotation Agent System for TaiWanese LOM (TW LOM)*. The 2nd International Conference on Theory and Application of Information Technology, Tainan Taiwan.

[14] Li, Juanzi, Fan, Qi'na, and Kuo, Zhang (2007). Keyword Extraction Based on TFIDF for Chinese News Document. *Wuhan University Journal of Natural Sciences, 12*(5), 917-921.

[15] Liu, Cheng-Chang, Yeh, Jen-Yuan, Ke, Hao-Ren, and Yang, Wei-Pang (2005). Concept Cluster Based News Document Summarization. *Proceedings of the 17th ROCLING Conference on Computational Linguistics and Speech Processing (ROCLING 2005),* 99-111.

[16] Liu, Chuan-han, Wang, Yong-cheng, Zheng, Fei, and Liu, De-rong (2007). Using LSA and Text Segmentation to Improve Automatic Chinese Dialogue Text Summarization. *Journal of Zhejiang University—Science A, 8*(1), 79-87.

[17] Luhn, Hans P. (1958). *The Automatic Creation of Literature Abstracts. IBM Journal of Research and Development, 2*(2), 159-165.

[18] Salton, Gerard, Singhal, Amit, Mitra, Mandar, and Buckley Chris (1997). Automatic Text Structuring and Summarization. *Information Processing & Management,* Volume 33, Number 2, 193-207, 193-207.

[19] Shen, Dou, Sun, Jian-Tao, Li, Hua, Yang, Qiang, and Chen Zheng (2007). Document Summarization using Conditional Random Fields. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007),* 2862-2867.

[20] Silber, H. Gregory, and McCoy, Kathleen F. (2000). Efficient Text Summarization Using Lexical Chains. *Proceedings of the 5th international conference on intelligent user interfaces,* 252-255.

[21] Tseng, Yuen-Hsien (2004). Automatic Summarizing Text Messages from Chinese News. *ROCLING XVI: Conference on Computational Linguistics and Speech Processing*, Taipei, Taiwan, 177-189.

[22] Wang, Jian-Hui, Zhou, Shui-Geng, and Hu, Yun-Fa (2003). Sentences Clustering Based Automatic Summarization. *International Conference on Machine Learning and Cybernetics, 1*, 57-62.

[23] Yeh, Jen-Yuan, Ke, Hao-Ren, and Yang, Wei-Pang (2002). Chinese Text Summarization Using a Trainable Summarizer and Latent Semantic Analysis. *Proceedings of the 5th International Conference on Asian Digital Libraries: Digital Libraries: People, Knowledge, and Technology, 2555*, 76-87.

[24] Zhang, Dell, and Dong, Yisheng (2004). Semantic, Hierarchical, Online Clustering of Web Search Results. *Proceedings of the 6th Asia Pacific Web Conference (APWEB),* 69-78

[25] ADL SCORM standard: http://www.adlnet.gov/downloads/AuthNotReqd.aspx?FileName=SCORM.2004.3ED.DocSuite.zip&ID=237

[26] Chinese Knowledge and Information Processing (CKIP) Chinese Parser: http://rocling.iis.sinica.edu.tw/CKIP/

[27] IEEE LOM standard: http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

[28] Sciscape: http://www.sciscape.org/list_by_cat.php?cat=%A6a%B2y%AC%EC%BE%C7&sn=0

[29] Self-Organizing Maps (SOMs) : http://www.improvedoutcomes.com/docs/WebSiteDocs/Introduction/Tutorials/Tutorial_4_Self_Organizing_Maps_(SOMs)/Tutorial_4__Introduction.htm

[30] TIPSTER Text Program: http://www-nlpir.nist.gov/related_projects/tipster/